APPENDIX D

SPECS SURVEILLANCE FACTOR DEFINITIONS

1. The following lists the surveillance or risk factors from SPECS Environmental Factor input screens, in order. Each is listed by the screen label, with the 4 character REVIC acronym for reference. It is then followed by the definition for determination of the factor levels. At the end of the paragraph the selections for input are summarized. The SPECS exclusive or Non-REVIC factors are the Software Engineering Institute (SEI) CMM level and Program Visibility (ACAT code).

2. **Analysts Capability (ACAP)** measures the capability of the analysts performing the following: development and validation of requirements, preliminary design SPECS and test plans; consultant effort during detailed design and code and unit test phases; and heavy participation during the integration and test phase. ACAP is broken into three components. They are: previous program performance, education, and communication. Keep in mind that this parameter deals only with the analyst team that will actually be working on the project. The efforts of the programming team are considered separately in PCAP, which is the next factor.

a. One indicator of an analyst teams capability is their previous program performance. By this we mean how effective has this same team been in past projects. Items to be considered include:

(1) Did the end product perform as required with respect to speed, reliability, and functionality?
(2) Was it completed in the projected timeframe?
(3) Was the project brought in within the projected budget?

b. Another predictor of the analysts teams ability is the level and type of their formal education. Factors to be considered include:

(1) Level of education - PHD, MS, MBA, etc.
(2) Relevant areas of study -S/W engineering, math, etc.

c. Effective communication and cooperation is vital to an efficient software development team effort. Considerations here include structure of the team, formal communication policies, and location of team. Examples might include:

(1) A highly rated team - centralized team; clear cut lines of communication.
(2) A medium rated team - some team members not co-located; lines of communication established but not always observed.
(3) A low rated team - team members scattered; little structured communication.

d. ACAP measures the capability of the analyst team doing the following: development and validation of requirements, preliminary design SPECS and test plans; consultant effort during detailed design and code and unit test phases; and heavy participation during the integration and test phase. All of these factors should be summarized into the ACAP rating as follows:

(1) Very Low - 15th Percentile Team
(2) Low - 35th Percentile
(3) Nominal - 55th Percentile

3. **Programmer Capability (PCAP)** measures the capability of the programmers performing the detailed CSCI/CSC design during the critical design phase of the contract and writing and testing the physical code during the coding and integration testing phases. We have broken down the analysis of the PCAP into three components. They are: previous program performance, education, and communication. Keep in mind that this parameter deals only with the programming team that will actually be working on the project. The efforts of the analyst team are considered separately. One indicator of programming teams

capability is their previous program performance. By this we mean how effective has this same team been in past projects.

a. Items to be considered include:

(1) Did the end product perform as required with respect to speed, reliability, and functionality?
(2) Was it completed in the projected timeframe?
(3) Was the project brought in within the projected budget?

b. Another predictor of the programming teams ability is the level and type of their formal education. Factors to be considered include:

(1) Level of education - PHD, MS, MBA, etc.
(2) Relevant areas of study – S/W engineering, math, etc.

c. Effective communication and cooperation is vital to an efficient software development team effort. Considerations here include structure of team, formal communication policies, and location of team.

(1) High - centralized team; clear cut lines of communication.
(2) Medium - some team members not co-located; lines of communication established but not always observed.
(3) Low - team members scattered; little structured communication.

d. These factors should be combined into the same Percentile ranges as used in ACAP, above. PCAP measures the capability of the programmers performing the detailed module design during the critical design phase and the writing and test of the physical code during the coding and integration testing phases.

4. **Applications Experience (AEXP)** rates the design and development team with respect to their level of experience working on similar software projects. The criteria are explicit with a specific time period for each rating. An important point must be considered when determining the depth of experience. The experience only applies when the design team worked on similar projects. Similarity does not mean working on the same equipment or even in the same language. Similarity means working on projects of like difficulty. Projects designing software that directly controls the hardware (operating systems, compilers, and embedded systems) are examples of difficult design experience. Conversely, applications programs used in a batch mode are of low difficulty. Experience on projects of like difficulty is the key for AEXP.

a. Before we look at Applications experience, a few words about consolidating the experience of multiple team members into a discrete number. While subjective, there are some things to keep in mind. Most of the time using a simple average of the individuals experience is the best approach. When the experience of one member of the group is highly skewed from the rest, however, using the mode of the individuals experience is preferable.

b. Whenever a single individual exerts a disproportionate amount of influence on the process in question (i.e. a noted authority or expert consultant) it would be wise to weight their experience more heavily into the group total. Considering the experience of the design and development team with projects of like degree of difficulty, choose the rating that best applies. Remember, the team consists of both systems analysts and management personnel. AEXP rates the design and development team with respect to their level of experience working on projects of like scope and difficulty.

(1) VL - Less than four months relevant experience
(2) LO - Five months to two years relevant experience
(3) NM - Two years or more relevant experience

5. **Virtual Machine Experience (VEXP)** measures the design (analysts) and programmer's experience with the hardware and software of the host and target computers. Host computer is the machine the software will be developed on. Target computer is that machine the software will run on. The software of the host and target computers refers to the system software (the operating system), development editors, formatters, compilers, linkers, and other computer aided software tools used by the development team. Use of a high order language (HOL) affects VEXP. An HOL is a language that is largely machine independent, such as ADA. Using a HOL usually does not require a detailed knowledge of the hardware or systems software. The length of familiarity is also a critical indicator of VEXP. It is important to understand that merely because a particular programming language is classified as a HOL does not mean it is used in a machine independent fashion. For example, the C language may be used at different levels. It may be used as a HOL where issuing the print command takes care of all of the functions necessary to print independently of the machine being used. At a far lower level C may be used to move the address of a particular piece of information to a particular output port. His action is totally machine dependent. So to determine the appropriate value of VEXP, we must find out if a HOL is being used in a machine independent fashion.

(1) VL - The team has no experience with the hardware and software required
(2) LO - Less than 6 months experience
(3) NM - Greater than 6 months experience

6. **Programming Language Experience (LEXP)** measures the design (analysts) and programmer's team experience with the programming language that will be used to design the software. The input criteria are straight forward measures of length of experience in years, if the contractors proficiency and the development language are the same. If they are not, one additional question should be considered. Is the language of the contractor's experience similar to that required to build the software? Similarity means that a portion of the proficiency in one language can be transferred.

a. The general REVIC rule is to rate as:

(1) VL - less than one month relevant experience
(2) LO - one month to eleven months of relevant experience
(3) NM - 12 or more months of relevant experience

7. **Execution Time Constraint (TIME)** measures the approximate percentage of the available CPU (Central Processing Unit) execution time that will be used by the software. Stated in other words, TIME deals with the throughput of the target computer. Throughput is the number of operations per second that a particular CPU is capable of performing. TIME is an attempt to rate how much of the target computers throughput will be used by the project software. Although this throughput is quantifiable, a discussion of its calculation is beyond the scope of this document. (See notes from the embedded software class for a further discussion of the calculation of throughput.) For use in SPECS you have several sources of information for this parameter.

a. Check the proposal or contract first. Often these documents will address the specific time constraint
requirements, if any.

b. Do some comparisons by analogy. See if the contractor has produced similar projects using the same CPU. This will give you an idea of the likelihood of time constraints, if any. Normally, the only projects that involve execution time constraints are real time applications or online weapons systems (i.e. target acquisition radar).

c. The most valuable source of information will be the engineering people at the procuring office or buying command. They should be able to shed some light on both the design to specifications for this particular project as well as comparisons by analogy.

(1) NM - Utilization less than 70%
(2) HI - Utilization is between 70% and 85%
(3) VH - Utilization is between 85% and 90%
(4) XH - Utilization is greater than 90%

8. **Main Storage Constraint (STOR)** measures the limitations on the software due to constraints on memory in the target computer. Memory is that random access storage which is directly accessible to the CPU or I/O processor (i.e. core integrated circuit, or plated-wire storage). Not included are peripheral devices such as disks, drums, tape, or bubble storage. Memory limitations can be avoided by moving information to and from peripheral devices. Doing that eats up execution time, so there is a trade off between execution time and main storage constraints.

a. The same as for TIME, this rating should only be a factor for real time, embedded systems. There is a methodology for developing an independent estimate of storage constraints. The percent of storage utilization is calculated simply by I/T where I is the estimated storage requirement for the project software and T is the memory capacity of the target computer. The estimated storage requirement is computed by converting the estimated lines of source code to object code as follows:

(1) $I = [L(1+P(E-1))]/W$

b. Where I is the number of object instructions, L is the number of source lines of code, E is the expansion ratio of the High Order Language (HOL) used, P is the percent of code written using the HOL, and W is the word length of the target computer. In simple terms, the overall memory utilization, while the program, is in execution breaks down to the following ratings:

(1) NM - No memory constraints; either unlimited memory or less than 70% used
(2) HI - Utilization of 70 to 85%
(3) VH - Utilization of 85 to 95%
(4) XH - Utilization over 95%

9. **Virtual Machine Volatility (VIRT)** measures the amount of change the host and target machines are undergoing during the design and development. Machine here refers to the hardware and software that the project calls on to complete its task. Depending on the scope of the project, the virtual machines could include: the operating system, a database management system, and assemblers and compilers for languages in which the project is written. If this is being evaluated before development, the user must rely on history of similar projects as well as any changes anticipated by the PMO and contractor. Not all system changes impact the project development.

a. For example, if application programs are being produced in a HOL and the compiler, editor and formatter are unchanged, then a change in the host or target hardware would have little impact on the development. Changes to the host should be considered only if there is a solid reason. For example, if the change results in better turn around times or the opportunity to use CASE (Computer Assisted Software Engineering) tools, then it would be justified. Otherwise these should not be included in the VIRT evaluation.

b. The following is the SPECS rating guideline for VIRT. Count only those changes having a significant impact on development. Dr. Boehm (COCOMO) would define significant impact in terms of the percent of routines under development that are affected. Using this approach SPECS users could define a significant change as impacting roughly 5% of the routines under development.

(1) NM - One change expected every three months
(2) HI - One change every month
(3) VH - Several changes every month
(4) XH - Concurrent hardware development

10. **Requirements Volatility (RVOL)** is a PARM specific parameter that measures the amount of redesign and development that is the result of changes in customer specified requirements. Projects are contracted on the basis of requirements known at the time of negotiation. This factor is designed to account for the extra systems engineering and management effort required to evaluate the changed

requirements, estimate the impact, prepare the Engineering Change Proposal and change the software. Do not use this factor for building in a management reserve for risk as this will be addressed separately.

a. Use the following criteria to choose the best input for RVOL. The choice should be made based on the actual history of the project and thus should be straight forward.

(1) NM - Small noncritical redirections
(2) HI - Occasional moderate redirections
(3) VH - Frequent moderate or occasional major redirections
(4) XH - Frequent major redirections

11. **Software Development Mode (MODE)** is the mode of software development. In SPECS only two modes from REVIC have been made available:

a. Semidetached (SD): A combination of organic and embedded features. A program that contains organic criteria, but may get input from another computer, special peripheral devices, and may do specialized processing.

b. Embedded (EB): A program with considerable interfaces, new algorithms and/or extremely tight constraints, usually very large or complicated programs. Most of the Mil-SPEC systems for weapons or vehicle control or command and control would fit this definition.

12. **Product Reliability (RELY)** is a generalized expression of program reliability. As Dr. Boehm describes in his book, RELY is easy to define but very tough to quantify. RELY is the probability that the software performs its intended function over its next run. The problems in quantifying this stem from such things as defining "perform its intended function" and "choosing a random, representative set of test data that completely simulates real world experience".

a. For this reason the evaluation criteria for RELY is qualitative. The criteria we will use include the degree of loss a failure causes and the requirements of a Mil-Std type of development. Once we have established there is a requirement for development to a Mil-Std, we must establish the degree of impact that a software failure will have. Select from the following that which describes the effect of a failure:

(1) XH - Full Mil Std development with highest possible risk, including loss of many lives and/or possible damage to national security.
(2) VH - Full or tailored Mil Std development with possible loss of life.
(3) HI - Typically tailored Mil Std development where recovery of failure takes more than 40 manhours or financial loss is in excess of $10,000.
(4) NM - Recovery of failure takes from 3 - 40 Manhours or financial loss is in range of $1,000 to $10,000 and is typically a non Mil Std development.

13. **Data Base Size (DATA)** attempts to quantify the effort required to create a project that will manipulate and maintain a sizeable data base. This factor is evaluated by the following formula, DB/DSI. DB stands for the size of the data base in bytes. DSI is the number of deliverable source instructions that you have used as the estimate of the total size of the project.

(1) NM, HI, VH and XH are calculated automatically by factor help.

14. **Product Complexity (CPLX)** rates the complexity of the software to be developed. Software complexity is directly related to the type of processing required and the distance from the hardware. As Dr. Boehm points out in his book Software Cost Economics, although attempts have been made to develop metrics for CPLX, none has proven universally accepted. As a result the CPLX rating is done by analogy.

a. The ratings and their corresponding rules of thumb are:

(1) NM - Data processing, standard math and statistical routines
(2) HI - Some hardware input/output and advanced data routines and tools such as compilers
(3) VH - Real time applications; structured numerical analysis; partial differential equations; embedded systems
(4) XH - Extremely complex scientific processes such as signal processing; unstructured numerical analysis; analysis of noisy stochastic data

15. **Required Reuse (RUSE)** measures the extra effort needed to generalize software modules when they must be developed specifically for reuse in other software packages. Notice that reuse does not refer to a boilerplate statement concerning the use of ADA code. Only code designed for specific reuse should be considered in RUSE. Although not originally included in COCOMO, Ray Kile has included RUSE as a significant factor in REVIC. Interestingly, Dr.Boehm has also included this factor in the ADA version of COCOMO.

a. First, determine that there is a contractual requirement for reuse of code. Note, however, that this does not include the boiler plate statement regarding the reuse of ADA code. Choose the value that best reflects the reusability requirements:

(1) NM - No reuse required.
(2) HI - Reuse in single mission products i.e. space station or up to 30% of the modules are designed for reuse.
(3) VH - Reuse across single product line i.e. any space craft or up to 50% of the modules are designed for reuse.
(4) XH - Reuse in any application i.e. anything that flies or over 50% of the modules are designed for reuse.

16. **Capability Maturity Model** (CMM) refers to the Carnegie-Mellon SEI CMM level. The Capability Maturity Model Level assigns a risk factor based on the contractor's software process maturity as defined by the Software Engineering Institute's (SEI) CMM. If the contractor has performed or hired a contractor to perform a software process assessment, use their rating unless you have reason to challenge it. If the contractor is unaware of SEI and/or has not performed a software process assessment, rate the contractor at the initial (IN) level.

(1) IN - Level I Initial
(2) RP - Level 2 Repeatable
(3) DF - Level 3 Defined
(4) NM - Level 4 or 5 Managed/Optimized

17. **Use of Software Tools (TOOL)** rates the use of automated software tools in one of three categories from VL to NM.

a. Review the following list to determine the best description of the tools that may be in use:

(1) VL - Very few primitive tools.
(2) LO - Basic microprocessor tools; a basic assembler, linker and monitor with batch debugging aids.
(3) NM - Basic mini tools; HOL compiler, macro assembler, simple overlay linker, batch source editor, basic library and data base aids.

Use NM if extensive tools or integrated environments are used.

18. **Program Visibility (Acquisition Category, ACAT)** is another SPECS exclusive rating. It is tied to Program Visibility and is based primarily on Acquisition Category (ACAT) with monies expressed in

FY90 constant dollars. Any contract with a DX rating (per Defense Priorities and Allocation System, DPAS) will be rated at a minimum of E3.

a. Other considerations or customer concerns expressed may elevate visibility above the levels shown:

(1) El = ACAT I - An eventual expenditure of $300 million in RDT&E or $1.8 billion in procurement.
(2) E2 = ACAT II - An eventual expenditure of $115 million in RDT&E or $540 million procurement.
(3) E3 = ACAT III - Programs not meeting ACAT I or II that have been designated Category III by the DOD Component Acquisition Executive.
(4) E4 = ACAT IV - All other acquisition programs for which the milestone decision authority has been delegated to a level below that required for Category 111.

19. **Management Reserve for Risk (RISK)** is the parameter that provides for adding a percentage factor to account for varying levels of program risk. SPECS warns that this factor should only be used very carefully.

a. Review the following and select the appropriate response:

(1) VL - Very low program risk (ground systems)
(2) LO - Low program risk (Mil-Spec ground systems)
(3) NM - Medium program risk (unmanned airborne systems)
(4) HI - High program risk (manned airborne systems)
(5) VH - Very high program risk (unmanned space applications)
(6) XH - Extra high program risk (manned space applications)

20. **Required Schedule (SCED)** measures the effect that compressing or extending the schedule will have on the total manmonths. SPECS has used only the VL, LO and NM criteria for SCED:

(1) VL - Contract schedule less than .85 predicted by COCOMO/REVIC.
(2) LO - Contract schedule between .85 and .95 predicted by REVIC.
(3) NM - Contract schedule longer than .95 REVIC predicted schedule.